

47 supplying, [in succession] from said instruction[register] groups, using the predetermined location, said operand or [sequential instructions] instruction or both [of said certain of said instruction groups] to said central processing unit[; and  
[selecting an operand from said one of said instruction groups for use by said central processing unit].

#### REMARKS

Appreciation is expressed for the courteous and helpful telephone interview granted by the Examiner on January 29, 1998 and on February 2, 1998 in response to a proposal made by the Examiner on January 28, 1998 to add the limitations of claims 74 and 94 to independent claims 71 and 91, respectively, as well as to cancel claim 97 in order to place the application in condition for allowance. In the interview, Applicants' attorney and Mr. George Shaw, a technical representative of the assignee of this application, presented claim 71 similarly as amended above as a proposal that introduces many of the limitations of dependent claim 74 into that claim, but does not limit its scope to a MICROLOOP instruction. This version also differs somewhat from the version transmitted by facsimile in a proposed amendment on January 29, 1998 as a result of realizing that some of the language formerly moved from claim 74 to claim 71 applies only in the case of SKIP and MICROLOOP instructions, and to address the Examiner's concerns expressed in the February 2, 1998 interview. As a result, the language applicable only to those instructions has been left in amended dependent claims 72 and 74.

One of the unique characteristics of the claimed processor and processing method is the locating of operands or instructions by their position within the current group of instructions, or a characteristic we call "groupedness". This characteristic is represented in the disclosure by example in various instructions that are the subject of the dependent claims.

In the January 29, 1998 interview, the Examiner desired to know where this aspect of the invention was taught in the application as filed. Claims 71, 91 and 97 now more clearly define where instructions and operands of concern are located as "a predetermined position from a boundary of said instruction groups." For

reference, the examples in the specification that support the claims as amended are cited below.

Page 29, line 9, to page 30, line 9, discuss the SKIP instruction. Specifically, from lines 30-34 on page 29 "As shown, the SKIP operation is implemented by resetting the 2-bit microinstruction counter 180 to zero on line 422 and simultaneously latching the next instruction group into register 108." The result of resetting the counter is to begin executing at the beginning boundary of the next instruction group.

Page 30, lines 10 to 36, discuss the MICROLOOP instruction. Specifically, lines 25-28 "If not zero, the LOOP COUNTER 92 is decremented and the 2 bit microinstruction counter [180] is cleared, causing the preceding instruction in the instruction register to execute again." The result of clearing the counter is to begin executing at the beginning boundary of the current instruction group.

Page 33, lines 6 to 28 discuss variable width operands associated with JUMP-type (branch) or IMMEDIATE instructions. Specifically, from lines 15-19 "This magic is possible because operands must be right justified in the instruction register. This means that the least significant bit of the operand is always located in the least significant bit of the instruction register." Thus the operand is located from the end boundary of the current instruction group. Page 42, line 6 to page 45, line 20 supplies a detailed discussion of how branches and calls and the associated variable width operands are encoded in binary.

Page 41, line 36, and page 42, line 1, indicate "Branches and calls are made to 32-bit word boundaries." These boundaries correspond to instruction group boundaries and thus branches and calls also reference instructions that are located at a predetermined position from a boundary of an instruction group.

Page 55, lines 5-25, discuss FETCH-VIA-PC. Specifically, from lines 5-14, "Fetch the 32-bit memory content pointed to by the Program Counter and push it onto the Parameter Stack. After fetching, increment the most significant 30 bits of the Program Counter to point to the next 32-bit word address. \*NOTE When this instruction executes, the PC is pointing to the memory location following the instruction. The effect is of loading a 32-bit immediate operand." Thus the operand, the 32-bit memory content fetched, is located at a predetermined position from a

boundary of an instruction group, that is, the next instruction group, and fills an entire instruction group.

Page 59, lines 6-33 discuss LOAD-SHORT-LITERAL. Specifically, lines 9-12: "This instruction requires that the 8-bits to be pushed reside in the last byte of a 4-byte instruction group. The instruction op-code loading the literal may reside in ANY of the other three bytes in the instruction group." Thus, the operand for this instruction is located a predetermined position from the end boundary of the current instruction group.

The Examiner's concern of missing structure in claims 71 and 91 are believed to be resolved. This required extensive reorganization of the claims to separately detail instruction access and operand access. From claim 71:

instruction decoding means having a means for generating a counter control signal and an operand control signal;

a counter that is connected to receive said counter control signal from said instruction decoding means;

operand selection means that is responsive to said operand control signal from said instruction decoding means;

instruction supplying means responsive to said counter to select said predetermined position, for supplying, in succession from said instruction register, said sequential instructions to said central processing unit;

said instruction supplying means being further responsive to said counter and said operand selection means, for selecting and supplying operands, from said predetermined position in said instruction groups, to said central processing unit;

said instruction decoding means providing said counter control signal and said operand control signal to cause said instruction supplying means to select from said instruction groups said operand or instruction or both associated with one of said instructions from said first of said instruction groups.

Similar text was added to claim 91.

The counter 180 and operand selection means (442, 440, 180) are described in the specification at page 33, and is shown in FIG 20 to be responsive to control signals provided by the instruction decoding means.

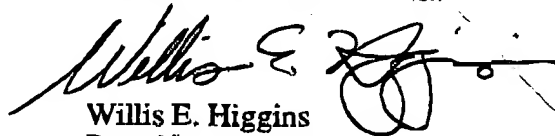
Similarly, the more specific recitation of locating the operand or instruction a predetermined position from a group boundary and a counting step and have been

added to claim 97. The dependent claims above have been changed as appropriate to correspond to the changes in their parent claims.

Based on the interview, these changes to the claims and the above remarks, this application is believed to be in condition for allowance, and allowance is solicited. The claims as amended define subject matter not taught or suggested in the prior art and are based on the teaching in the application as filed, as discussed above.

Respectfully submitted,

COOLEY GODWARD LLP

  
Willis E. Higgins  
Reg. No. 23,025

Five Palo Alto Square  
Palo Alto, CA 94306-2155  
Telephone: (650) 843-5145